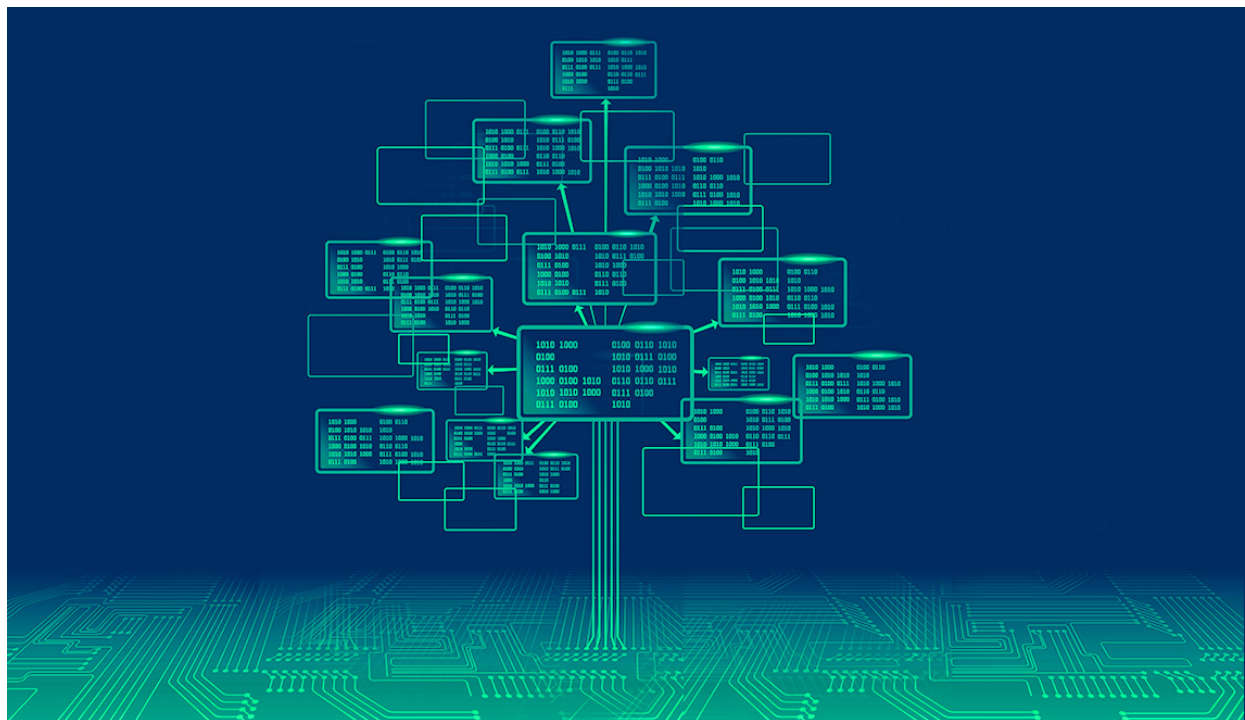


THE XORIENT BLOG

PRODUCT ENGINEERING ([HTTPS://WWW.XORIENT.COM/BLOG/CATEGORY/PRODUCT-ENGINEERING](https://www.xoriant.com/blog/category/product-engineering))

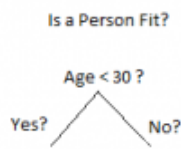
Decision Trees for Classification: A Machine Learning Algorithm

September 7, 2017 by Mayur Kulkarni



Introduction

Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split.



We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

Trees-modified-1.png)

An example of a decision tree can be explained using above binary tree. Let's say you want to predict whether a person is fit given their information like age, eating habit, and physical activity, etc. The decision nodes here are questions like 'What's the age?', 'Does he exercise?', 'Does he eat a lot of pizzas'? And the leaves, which are outcomes like either 'fit', or 'unfit'. In this case this was a binary classification problem (a yes no type problem).

There are two main types of Decision Trees:

1. Classification trees (Yes/No types)

What we've seen above is an example of classification tree, where the outcome was a variable like 'fit' or 'unfit'. Here the decision variable is **Categorical**.

2. Regression trees (Continuous data types)

Here the decision or the outcome variable is **Continuous**, e.g. a number like 123.

Working

Now that we know what a Decision Tree is, we'll see how it works internally. There are many algorithms out there which construct Decision Trees, but one of the best is called as **ID3 Algorithm**. ID3 Stands for **Iterative Dichotomiser 3**.

Before discussing the ID3 algorithm, we'll go through few definitions.

Entropy

Entropy, also called as Shannon Entropy is denoted by $H(S)$ for a finite set S , is the measure of the amount of uncertainty or randomness in data.

$$H(S) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$$

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-2.jpg>)

Intuitively, it tells us about the predictability of a certain event. Example, consider a coin toss whose probability of heads is 0.5 and probability of tails is 0.5. Here the entropy is the highest possible, since there's no way of determining what the outcome might be. Alternatively, consider a coin which has heads on both the sides, the entropy of such an event can be predicted perfectly since we know beforehand that it'll always be heads. In other words, this event has **no randomness** hence its entropy is zero.

In particular, lower values imply less uncertainty while higher values imply high uncertainty.

Information Gain

Information gain is also called as Kullback-Leibler divergence denoted by $IG(S,A)$ for a set S is the effective change in entropy after deciding on a particular attribute A . It measures the relative change in entropy with respect to the independent variables.

$$IG(S, A) = H(S) - H(S, A)$$

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-3.jpg>)



Alternatively,

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

where $IG(S, A)$ is the information gain by applying feature A . $H(S)$ is the Entropy of the entire set, while the second term calculates the Entropy after applying the feature A , where $P(x)$ is the probability of event x .

Let's understand this with the help of an example

Consider a piece of data collected over the course of 14 days where the features are Outlook, Temperature, Humidity, Wind and the outcome variable is whether Golf was played on the day. Now, our job is to build a predictive model which takes in above 4 parameters and predicts whether Golf will be played on the day. We'll build a decision tree to do that using **ID3 algorithm**.

Day	Outlook	Temperature	Humidity	Wind	Play Golf
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

ID3 Algorithm will perform following tasks recursively

1. Create root node for the tree
2. If all examples are positive, return leaf node 'positive'
3. Else if all examples are negative, return leaf node 'negative'
4. Calculate the entropy of current state $H(S)$
5. For each attribute, calculate the entropy with respect to the attribute 'x' denoted by $H(S, x)$
6. Select the attribute which has maximum value of $IG(S, x)$
7. Remove the attribute that offers highest IG from the set of attributes
8. Repeat until we run out of all attributes, or the decision tree has all leaf nodes.

Now we'll go ahead and grow the decision tree. The initial step is to calculate $H(S)$, the Entropy of the current state. In the above example, we can see in total there are 5 No's and 9 Yes's.



Yes	No	Total
-----	----	-------

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

$$\text{Entropy}(S) = -\left(\frac{9}{14}\right) \log_2 \left(\frac{9}{14}\right) - \left(\frac{5}{14}\right) \log_2 \left(\frac{5}{14}\right) = 0.940$$

[Trees-modified-5.jpg](#)

Remember that the Entropy is 0 if all members belong to the same class, and 1 when half of them belong to one class and other half belong to other class that is perfect randomness. Here it's 0.94 which means the distribution is fairly random.

Now the next step is to choose the attribute that gives us highest possible Information Gain which we'll choose as the root node.

Let's start with 'Wind'

$$IG(S, \text{Wind}) = H(S) - \sum_{i=0}^n P(x) * H(x)$$

[modified-6.jpg](#)

where 'x' are the possible values for an attribute. Here, attribute 'Wind' takes two possible values in the sample data, hence $x = \{\text{Weak}, \text{Strong}\}$

We'll have to calculate:

1. $H(S_{\text{weak}})$
2. $H(S_{\text{strong}})$
3. $P(S_{\text{weak}})$
4. $P(S_{\text{strong}})$
5. $H(S) = 0.94$ which we had already calculated in the previous example

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-77.jpg>) Amongst all the 14 examples we have

8 places where the wind is weak and 6 where the wind is Strong.

Wind = Weak	Wind = Strong	Total
8	6	14

$$P(S_{\text{weak}}) = \frac{\text{Number of Weak}}{\text{Total}} = \frac{8}{14}$$

$$P(S_{\text{strong}}) = \frac{\text{Number of Strong}}{\text{Total}} = \frac{6}{14}$$

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-77.jpg>)

[Trees-modified-8.jpg](#)

Now out of the 8 Weak examples, 6 of them were 'Yes' for Play Golf and 2 of them were 'No' for 'Play Golf'. So, we have,

$$Entropy(S_{weak}) = -\left(\frac{6}{8}\right)\log_2\left(\frac{6}{8}\right) - \left(\frac{2}{8}\right)\log_2\left(\frac{2}{8}\right)$$

[_ \(https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-](https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-)

We use cookies on this site to enhance your user experience

[\(https://www.xoriant.com/cookie-policy\).](https://www.xoriant.com/cookie-policy)

No, give me more info

OK, I agree

'No' for Play Golf.

$$Entropy(S_{strong}) = -\left(\frac{3}{6}\right)\log_2\left(\frac{3}{6}\right) - \left(\frac{3}{6}\right)\log_2\left(\frac{3}{6}\right) = 1.000$$

[_ \(https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-](https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-)

[Trees-modified-10.jpg](#)

Remember, here half items belong to one class while other half belong to other. Hence we have perfect randomness.

Now we have all the pieces required to calculate the Information Gain,

$$IG(S, Wind) = H(S) - \sum_{i=0}^n P(x) * H(x)$$

$$IG(S, Wind) = H(S) - P(S_{weak}) * H(S_{weak}) - P(S_{strong}) * H(S_{strong})$$

$$= 0.940 - \left(\frac{8}{14}\right)(0.811) - \left(\frac{6}{14}\right)(1.00)$$

$$= 0.048$$

[_ \(https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-](https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-)

[Trees-modified-11.jpg](#)

Which tells us the Information Gain by considering 'Wind' as the feature and give us information gain of **0.048**. Now we must similarly calculate the Information Gain for all the features.

$$IG(S, Outlook) = 0.246$$

$$IG(S, Temperature) = 0.029$$

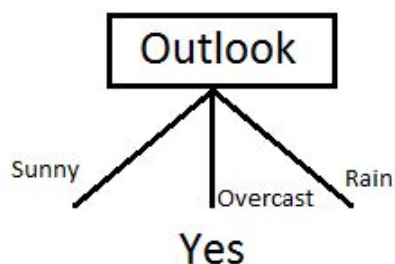
$$IG(S, Humidity) = 0.151$$

[_ \(https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-](https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-)

$$IG(S, Wind) = 0.048 \text{ (Previous example)}$$

[modified-12.jpg](#)

We can clearly see that $IG(S, Outlook)$ has the highest information gain of 0.246, **hence we chose Outlook attribute as the root node**. At this point, the decision tree looks like.



[_ \(https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-](https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-)

[modified-13.jpg](#)

Here we observe that whenever the outlook is Overcast, Play Golf is always 'Yes', it's no coincidence by any chance, the simple tree resulted because of **the highest information gain is given by the attribute Outlook**. ^

Now how do we proceed from this point? We can simply apply **recursion**, you might want to look at the algorithm steps

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

Next step would be computing $H(S_{\text{sunny}})$.

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-33.jpg>)

Table where the value of Outlook is Sunny looks like:

Temperature	Humidity	Wind	Play Golf
Hot	High	Weak	No
Hot	High	Strong	No
Mild	High	Weak	No
Cool	Normal	Weak	Yes
Mild	Normal	Strong	Yes

$$H(S_{\text{sunny}}) = \left(\frac{3}{5}\right) \log_2 \left(\frac{3}{5}\right) - \left(\frac{2}{5}\right) \log_2 \left(\frac{2}{5}\right) = 0.96$$

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-14.jpg>)

In the similar fashion, we compute the following values

$$IG(S_{\text{sunny}}, \text{Humidity}) = 0.96$$

$$IG(S_{\text{sunny}}, \text{Temperature}) = 0.57$$

$$IG(S_{\text{sunny}}, \text{Wind}) = 0.019$$

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-15.jpg>)

As we can see the **highest Information Gain is given by Humidity**. Proceeding in the same way with S_{rain}

(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-55.jpg>) will give us Wind as the one with highest information gain. The final Decision Tree looks something like this.

The final Decision Tree looks something like this.



(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision->

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

Code:

Let's see an example in Python

```

import pydotplus
from sklearn.datasets import load_iris
from sklearn import tree
from IPython.display import Image, display
__author__ = "Mayur Kulkarni <mayur.kulkarni@xoriant.com>"

def load_data_set():
    """
    Loads the iris data set

    :return:      data set instance
    """
    iris = load_iris()
    return iris

def train_model(iris):
    """
    Train decision tree classifier

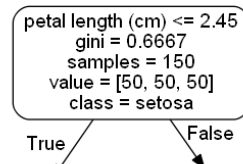
    :param iris:  iris data set instance
    :return:      classifier instance
    """
    clf = tree.DecisionTreeClassifier()
    clf = clf.fit(iris.data, iris.target)
    return clf

def display_image(clf, iris):
    """
    Displays the decision tree image

    :param clf:   classifier instance
    :param iris:  iris data set instance
    """
    dot_data = tree.export_graphviz(clf, out_file=None,
                                    feature_names=iris.feature_names,
                                    class_names=iris.target_names,
                                    filled=True, rounded=True)

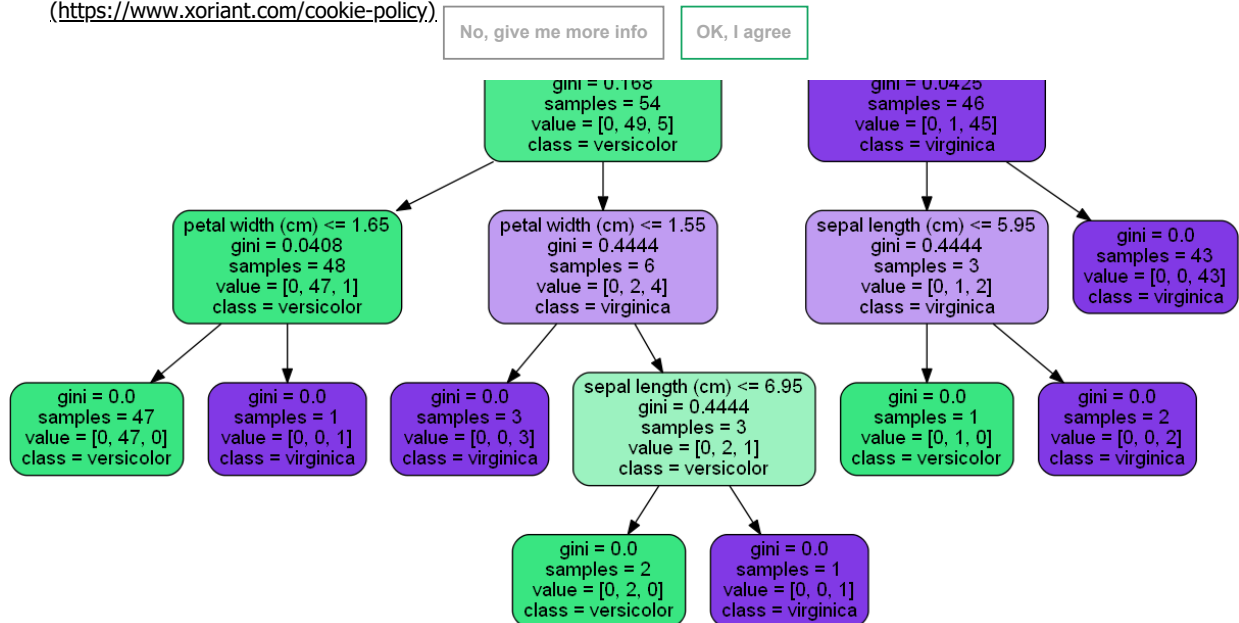
    graph = pydotplus.graph_from_dot_data(dot_data)
    display(Image(data=graph.create_png()))

if __name__ == '__main__':
    iris_data = load_iris()
    decision_tree_classifier = train_model(iris_data)
    display_image(clf=decision_tree_classifier, iris=iris_data)
  
```



We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).



(<https://www.xoriant.com/blog/wp-content/uploads/2017/08/Decision-Trees-modified-88.png>)

Conclusion:

Below is the summary of what we've studied in this blog:

1. Entropy to measure discriminatory power of an attribute for classification task. It defines the amount of randomness in attribute for classification task. Entropy is minimal means the attribute appears close to one class and have a good discriminatory power for classification
2. Information Gain to rank attribute for filtering at given node in the tree. The ranking is based on high information gain entropy in decreasing order.
3. The recursive ID3 algorithm that creates a decision tree.

References:

1. Information Gain: https://en.wikipedia.org/wiki/Information_gain_in_decision_trees
(https://en.wikipedia.org/wiki/Information_gain_in_decision_trees)
2. Entropy: [https://en.wikipedia.org/wiki/Entropy_\(information_theory\)](https://en.wikipedia.org/wiki/Entropy_(information_theory))
([https://en.wikipedia.org/wiki/Entropy_\(information_theory\)](https://en.wikipedia.org/wiki/Entropy_(information_theory)))
3. ID3: https://en.wikipedia.org/wiki/ID3_algorithm (https://en.wikipedia.org/wiki/ID3_algorithm)
4. Entropy YouTube video: https://www.youtube.com/watch?v=O_7IAqni7A (https://www.youtube.com/watch?v=O_7IAqni7A)
5. ID3 Example: <https://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm>
(<https://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm>)

Data Science (<https://www.xoriant.com/blog/tag/data-science>)



Decision Tree (<https://www.xoriant.com/blog/tag/decision-tree>)

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree



Mayur Kulkarni
Software Engineer

tt tt tt tt tt
p p p p p
s: s: s: s: s:
// // // // //
w w w w w
w w w w w
w. w. w. w. w.
x x x lin x
or or or k or
ia ia ia e ia
nt nt nt di nt
.c .c .c n. .c
o o o c o
m m m o m
lo lo lo /s lo
g/ g/ g/ h g/
pr pr pr ar pr
o o o e o
d d d Ar d
u u u tic u
ct ct ct le ct
- - - ?
n n n ini n
gi gi gi =t gi
n n n ru n

< **PREVIOUS POST** (<https://www.xoriant.com/blog/product-engineering/swagger-specification-code-generation.html>)

NEXT POST >

You may also like



(<https://www.xoriant.com/blog/big-data-analytics/augmented-analytics-accelerating-data-insights-into-actions.html>)

Augmented Analytics: Accelerating Data Insights into Actions (<https://www.xoriant.com/blog/big-data-analytics/augmented-analytics-accelerating-data-insights-into-actions.html>)

March 27, 2019

$$\begin{matrix} \bar{m} & \bar{m} & \bar{m} & \bar{m} & \bar{m} \\ l) & l) & l) & g/d & l) \end{matrix}$$


We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree



(<https://www.xoriant.com/blog/big-data-analytics/predictive-analytics-part-2.html>)

Predictive Analytics – PART 2 (<https://www.xoriant.com/blog/big-data-analytics/predictive-analytics-part-2.html>)

February 27, 2019



(<https://www.xoriant.com/blog/big-data-analytics/predictive-analytics-part-1.html>)

Predictive Analytics – PART 1 (<https://www.xoriant.com/blog/big-data-analytics/predictive-analytics-part-1.html>)

February 20, 2019

11 comments



We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree



September 12, 2017 at 7:13 am (<https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#comment-665587>)

Very intellectual and insightful.!!

REPLY ([HTTPS://WWW.XORIENT.COM/BLOG/PRODUCT-ENGINEERING/DECISION-TREES-MACHINE-LEARNING-ALGORITHM.HTML](https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html)?
REPLYTOCOM=665837#RESPOND)

Mayur Kulkarni



September 13, 2017 at 10:50 am (<https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#comment-665837>)

Glad that you liked our blog.

REPLY ([HTTPS://WWW.XORIENT.COM/BLOG/PRODUCT-ENGINEERING/DECISION-TREES-MACHINE-LEARNING-ALGORITHM.HTML](https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html)?
REPLYTOCOM=665981#RESPOND)

KRISHNA KADABUR



October 3, 2017 at 5:33 pm (<https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#comment-665981>)

nice article

REPLY ([HTTPS://WWW.XORIENT.COM/BLOG/PRODUCT-ENGINEERING/DECISION-TREES-MACHINE-LEARNING-ALGORITHM.HTML](https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html)?
REPLYTOCOM=669496#RESPOND)

priya



March 17, 2018 at 9:55 am (<https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#comment-669496>)

Hi,

I have read your Datascience blog.It's very attractive and impressive.Very useful to me, I like it your blog
Thanks for sharing this blog.....

Thank you

priya

Pingback: [Catboost – What's your data? – Medium - Coding Videos \(https://codingvideos.net/catboost-whats-your-data-medium/\)](https://codingvideos.net/catboost-whats-your-data-medium/)

Pingback: [An introductory guide to machine learning - Apro Software \(https://www.apro.3dize.com/an-introductory-guide-to-machine-learning/\)](https://www.apro.3dize.com/an-introductory-guide-to-machine-learning/)

Pingback: [Is machine learning entirely based on decision tree? | Physics Forums \(https://www.physicsforums.com/threads/is-machine-learning-based-on-decision-tree/\)](https://www.physicsforums.com/threads/is-machine-learning-based-on-decision-tree/)

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

REPLY (HTTPS://WWW.XORIENT.COM/BLOG/PRODUCT-ENGINEERING/DECISION-TREES-MACHINE-LEARNING-ALGORITHM.HTML?

REPLYTOCOM=696590#RESPOND)

Suhaimi Ariffin



April 16, 2019 at 12:24 pm (<https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#comment-696590>)

I like how you use math to model a decision tree. Thanks

REPLY (HTTPS://WWW.XORIENT.COM/BLOG/PRODUCT-ENGINEERING/DECISION-TREES-MACHINE-LEARNING-ALGORITHM.HTML?

REPLYTOCOM=697648#RESPOND)

priya



May 8, 2019 at 1:50 pm (<https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#comment-697648>)

Good Post! Thank you so much for sharing this pretty post, it was so good to read and useful to improve my knowledge as updated one, keep blogging.

Leave a Reply

Your email address will not be published. Required fields are marked *

Name*

Email*

Website

☐

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more infoOK, I agree

POST COMMENT

SUBSCRIBE TO OUR BLOG

Name *

Email *

☐ By submitting your information to Xoriant, you agree to our privacy (/privacy-policy) and cookie (/cookie-policy) policies.

SUBSCRIBE

Looking for something?

SEARCH

CATEGORIES

- Agile (<https://www.xoriant.com/blog/category/agile>) (7)
- Big Data and Analytics (<https://www.xoriant.com/blog/category/big-data-analytics>) (15)
- Cloud and Infrastructure (<https://www.xoriant.com/blog/category/cloud-infrastructure>) (4)
- Digital (<https://www.xoriant.com/blog/category/digital>) (1)
- Internet of Things (<https://www.xoriant.com/blog/category/internet-of-things>) (2)
- Mobility (<https://www.xoriant.com/blog/category/mobility>) (23)
- Product Engineering (<https://www.xoriant.com/blog/category/product-engineering>) (47)

TAGS

Adobe (<https://www.xoriant.com/blog/tag/adobe>)

^

We use cookies on this site to enhance your user experience

No, give me more info

OK, I agree

automation (<https://www.xoriant.com/blog/tag/automation>)

BI (<https://www.xoriant.com/blog/tag/bi>)

business intelligence (<https://www.xoriant.com/blog/tag/business-intelligence>)

Cloud (<https://www.xoriant.com/blog/tag/cloud>)

cloud computing (<https://www.xoriant.com/blog/tag/cloud-computing>)

Google (<https://www.xoriant.com/blog/tag/google>)

HTML (<https://www.xoriant.com/blog/tag/html>)

Internet of things (<https://www.xoriant.com/blog/tag/internet-of-things-2>)

Java (<https://www.xoriant.com/blog/tag/java>)

javascript frameworks (<https://www.xoriant.com/blog/tag/javascript-frameworks>)

microservices (<https://www.xoriant.com/blog/tag/microservices>)

Mobile application development (<https://www.xoriant.com/blog/tag/mobile-application-development-2>)

MySQL (<https://www.xoriant.com/blog/tag/mysql>)

Open Source (<https://www.xoriant.com/blog/tag/open-source>)

Product Engineering (<https://www.xoriant.com/blog/tag/product-engineering>)

QA (<https://www.xoriant.com/blog/tag/qa>)

R (<https://www.xoriant.com/blog/tag/r>)

software development (<https://www.xoriant.com/blog/tag/software-development>)

Software Testing (<https://www.xoriant.com/blog/tag/software-testing>)

Swagger (<https://www.xoriant.com/blog/tag/swagger>)

Twitter (<https://twitter.com/xoriant>)

We use cookies on this site to enhance your user experience

(<https://www.xoriant.com/cookie-policy>).

No, give me more info

OK, I agree

Web Application Development (<https://www.xoriant.com/blog/tag/web-application-development>)

Web Development (<https://www.xoriant.com/blog/tag/web-development>)

Windows (<https://www.xoriant.com/blog/tag/windows>)

1248 Reamwood Avenue, Sunnyvale, CA 94089 | Tel:+1 408 743 4400 (tel:+14087434400) |

info@xoriant.com (<mailto:info@xoriant.com>)

([http://](http://www.youtube.com/channel/UCU0)

[www.y](http://www.youtube.com/channel/UCU0)

[outube](http://www.youtube.com/channel/UCU0)

[.com/c](http://www.youtube.com/channel/UCU0)

[hannel](http://www.youtube.com/channel/UCU0)

[/UCU0](http://www.youtube.com/channel/UCU0) ([http://](http://www.youtube.com/channel/UCU0)

([http://l](http://www.linkedin.com/company/xoriant) ([https:/](https://www.linkedin.com/company/xoriant) c158x [www.sl](http://www.linkedin.com/company/xoriant)

([http://f](http://www.facebook.com/xoriant) [inkedi](http://www.facebook.com/xoriant) [/plus.g](https://www.facebook.com/xoriant) CKKX [idesha](https://www.facebook.com/xoriant)

acebo ([http://t](http://twitter.com/xoriant) [n.com/](https://twitter.com/xoriant) oogle. COqE [re.net/](https://twitter.com/xoriant)

ok.co [witter.c](https://twitter.com/xoriant) compa [com/+](https://twitter.com/xoriant) Fwi7R [xoriant](https://twitter.com/xoriant)

m/Xori [om/xor](https://twitter.com/xoriant) nies/1 [xoriant](https://twitter.com/xoriant) 3A/fee [corpor](https://twitter.com/xoriant)

ant) [iant\)](https://twitter.com/xoriant) 66996) [/posts\)](https://twitter.com/xoriant) d) [ation\)](https://twitter.com/xoriant)

© 2018 Xoriant Corporation. All Rights Reserved. | [Terms of Use \(/terms-use\)](/terms-use) | [Privacy Policy \(/privacy-policy\)](/privacy-policy)
| [Cookie Policy \(/cookie-policy\)](/cookie-policy) | [Sitemap \(/site-map\)](/site-map)